Une approche par apprentissage basée sur des modèles linguistiques

Omar Nouali, Alain Régnier, Philippe Blache

Classification de courriers électroniques

Une approche par apprentissage basée sur des modèles linguistiques

Omar Nouali[*,**] **— Alain Regnier**[**] **— Philippe Blache**[**]

[*] *Laboratoire de logiciels de base, CE.R.I.S.T.,*
Rue des 3 frères Aïssiou, Ben Aknoun, Alger, Algérie
onouali@mail.cerist.dz

[**] *LPL- Université de Provence,*
29, Av. Robert Schuman, F-13621 Aix-en-Provence, France
(regnier, pb)@lpl.univ-aix.fr

RÉSUMÉ. *Nous proposons une double amélioration des systèmes de filtrage de courriels existants. D'une part, en utilisant une méthode d'apprentissage automatique permettant à un système de filtrage d'élaborer des profils utilisateur. D'autre part, nous utilisons un ensemble de connaissances linguistiques sous forme de modèles réduits issues de modèles linguistiques de textes. Dans ce contexte, nous cherchons à évaluer si l'utilisation de connaissances et de traitements linguistiques peut améliorer les performances d'un système de filtrage. En effet, nous utilisons, au-delà des caractéristiques lexicales, un ensemble d'indicateurs sur le message portant sur la structure et le contenu. Ces connaissances sont indépendantes du domaine d'application et la fiabilité repose sur l'opération d'apprentissage. Pour tenter de statuer sur la faisabilité de notre approche et d'évaluer son efficacité, nous l'avons expérimenté sur un corpus de 1 200 messages. Nous présentons les résultats d'un ensemble d'expériences d'évaluation.*

ABSTRACT. *We propose a two-fold improvement to the existing e-mail filtering systems : firstly, by using an automatic learning method which will allow the filtering system to create user profiles. Secondly, we use a set of linguistic information in the form of reduced models, based on linguistic models of texts. In this area we aim to evaluate if using linguistic information and analysis can improve the performance of a filtering system. Indeed, as well as using lexical characteristics, we use a range of indicators based on structure and content of the messages. This information is independent to the application domain and reliability depends on the learning operation. In order to evaluate the feasibility of our approach and its reliability, we have experimented with a corpus of 1200 messages. We present here the results of a set of evaluation experiments.*

MOTS-CLÉS : *filtrage d'information, apprentissage automatique, modèles linguistiques réduits.*

KEYWORDS : *information filtering, machine learning, small-scale linguistic models.*

RSTI - RIA. Volume 19 – n° 6/2005, pages 883 à 910

1. Introduction

Les messageries de courriers électroniques proposent des systèmes de sélection de courriers. Cette sélection est une classification basée sur une propriété lexicale, la présence ou l'absence de mots-clés que l'utilisateur doit indiquer au logiciel. La présence ou l'absence de ces mots-clés en fonction des divers champs du courrier entrant dirige le courriel vers un dossier approprié contenant tous les courriels qui partagent cette (ces) même(s) propriété(s).

Le problème avec ces systèmes est que, d'une part, ils ne sont pas très précis car l'aspect sémantique est négligé, et, d'autre part, la nature des messages varie au cours du temps, ce qui nécessite une mise à jour fréquente des propriétés lexicales. Ces systèmes enregistrent donc des lacunes ou faiblesses sur l'efficacité du filtrage.

Pour améliorer ces systèmes, notre motivation a été d'explorer le potentiel des techniques de plusieurs domaines : le premier a trait au domaine de l'apprentissage automatique, passage obligé dans la conception d'un système de filtrage automatique d'information. Nous proposons donc une solution évolutive permettant au système d'apprendre à partir de données, d'exploiter ces connaissances et de s'adapter à la nature des courriers dans le temps. Le deuxième a trait à la nécessité d'utiliser des ressources ou traitements linguistiques. Dans ce contexte, nous voulons montrer que l'utilisation de connaissances et de traitements linguistiques peut améliorer les performances d'un système de filtrage. En effet, nous proposons un ensemble de connaissances linguistiques sous forme de modèles réduits (issues de modèles linguistiques de textes). Il s'agit d'un ensemble d'indicateurs sur le texte (portant sur la structure et le contenu du message). Un message est soumis à un processus d'analyse automatique, permettant de lui associer un ensemble de termes et de propriétés linguistiques, qui sert à le caractériser et à le situer par rapport aux autres. Ces connaissances sont indépendantes du domaine d'application, nous les avons classées en plusieurs niveaux : matériel, énonciatif, structurel et syntaxique. Dans le cadre de ce travail, nous ne cherchons pas à faire une analyse complète et profonde du contenu des messages, mais plutôt une analyse partielle utilisant plusieurs niveaux d'analyse dégageant des propriétés linguistiques qui devraient permettre de distinguer les différents types de messages et de classer ensuite les nouveaux messages.

Dans cet article, nous nous intéressons à quelques types génériques de messages bien particuliers : les messages personnels, professionnels et les messages indésirables (appelés *Spam*) qui continuent à polluer nos boîtes de courriels de façon croissante. Nous présentons à la fin les résultats d'un ensemble d'expériences d'évaluation.

2. Travaux antérieurs

L'analyse automatique de textes en langage naturel est en plein essor mais reste difficile. L'analyse de toutes les informations présentes dans un texte est un processus très complexe car il fait intervenir de nombreux paramètres. Les approches d'analyse de textes oscillent entre des analyses globales portant sur le texte dans son intégralité et des analyses beaucoup plus locales, répondant à des besoins particuliers.

L'approche globale est très ambitieuse, elle repose sur une analyse linguistique du texte dans son ensemble et vise une analyse profonde et exhaustive du texte.

L'approche locale, radicalement différente de l'approche globale, repose sur une analyse purement locale : l'objectif n'est plus une analyse extensive, seule une partie minime du texte nécessite une analyse approfondie.

Une application intéressante de l'analyse automatique de textes est l'élaboration des modèles qui permettent d'identifier et de classer les textes. Diverses études en linguistique informatique ont proposé des méthodes de classification de textes. Une grande majorité de ces travaux utilise la cooccurrence lexicale comme base de leur classification (Sebastiani, 1999). Les méthodes vectorielles (Salton *et al.,* 1975) peuvent être considérées comme des dérivées de ce principe puisqu'elles utilisent la cooccurrence pour construire leurs vecteurs.

D'autres approches ont recours aux relations de sémantique lexicale. Celles-ci utilisent des ressources lexicales (thésaurus, dictionnaires…) telles que Wordnet (Miller, 1990 ; Junker et Abecker, 1997). D'autres études ont également été menées en vue d'exploiter des informations de plus haut niveau que le mot. Nous citons les travaux qui intègrent des séquences de mots en accord avec une grammaire (Lewis, 1992) ou purement statistiques (Amini, 2001 ; Caropreso *et al.,* 2001).

Les travaux inspirés par les études de Douglas Biber (1988), de Bronckart (Bronckart *et al.,* 1985) et de Habert (2000) dépassent le cadre de la simple cooccurrence lexicale. Par exemple, Biber identifie 67 propriétés de divers ordres pour classer les textes. Ces propriétés sont d'ordre syntaxique (temps verbaux, présence d'auxiliaires, passivation, nominalisations…) mais aussi sémantiques (classes d'adverbes, types de modalités). Copeck *et al.* (2000) ajoutent à un ensemble de propriétés syntaxiques et sémantiques, des propriétés d'ordre pragmatique telles que la présence d'une introduction ou l'utilisation de conventions.

Dans ces études les propriétés utilisées ne sont pas issues d'un modèle linguistique. Elles sont collectées sur la base d'observations directes ou de travaux antérieurs. Chez Biber les propriétés linguistiques étaient sélectionnées sur la base d'études sociolinguistiques orientées pour la plupart vers la distinction entre les productions orales et écrites. Chez Copeck, les propriétés sont issues de l'introspection des analystes, de l'observation sur corpus, de la comparaison entre

textes et de collectes effectuées sur des travaux antérieurs. Les groupements de ces propriétés en différents niveaux linguistiques sont postérieurs à leur collecte : on énumère dans un premier temps des propriétés hétérogènes qui ont pour but de distinguer les différents types de textes. De plus des différences existent entre les études en fonction :

– de la nature du corpus utilisé (annoté ou non, catégorisé ou non, volumineux ou non),

– des moyens mis en œuvre (manuel ou automatique),

– du but recherché (catégoriser, dégager un seul type pour en définir les traits essentiels, ou classifier pour observer des regroupements),

– du volume de données à traiter (ce qui permet un traitement coûteux ou non).

Dans notre cas, l'existence de corpus faisant défaut, on part de propriétés issues de modèles linguistiques des textes, la fiabilité du système repose sur l'apprentissage automatique (performances de l'apprentissage).

La démarche que nous avons choisi de suivre est de proposer d'abord une catégorisation dans un ensemble de catégories prédéterminées. Ce n'est que dans un deuxième temps que l'on effectue une classification (clustering), les classes ne sont pas connues à l'avance (Bellot et El-Bèze, 2001). Pour cela nous utilisons des modèles linguistiques réduits auxquels on associe des propriétés linguistiques. Ces propriétés permettent de dégager différents types de texte associés à des classes de courriels génériques qui serviront dans un deuxième temps à construire par apprentissage des classes spécifiques à l'utilisateur.

3. Architecture globale du système

Le système est composé des principaux modules suivants :

– un module de prétraitement qui détermine la langue de chaque message et le prépare aux différentes étapes ultérieures de l'analyse en sélectionnant les connaissances nécessaires ;

– un analyseur linguistique qui analyse les messages et délivre en sortie une représentation conceptuelle associée. Il utilise un ensemble de connaissances linguistiques de base sous forme de modèles réduits ;

– un module de classification et de filtrage qui permet de comparer un nouveau message avec les différents profils de l'utilisateur. La connaissance du système est modélisée par un réseau de neurones ;

– un module d'apprentissage qui permet d'améliorer l'efficacité et les performances du système. Il lui permet donc de s'adapter à l'évolution de l'environnement.

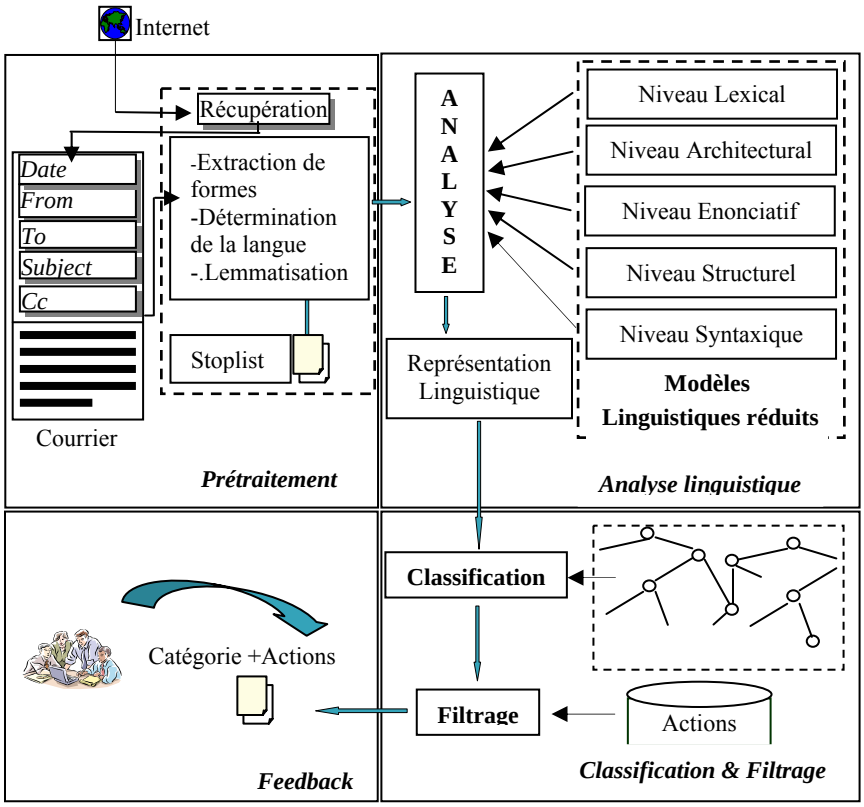

Figure 1. *Architecture globale du système*

3.1. *Prétraitement*

En premier, un module de ***prétraitement*** est lancé pour préparer les messages récupérés de la boîte de courriels, aux différentes étapes ultérieures de l'analyse. Il consiste à isoler les différents champs et à identifier la langue de chaque message parmi deux actuellement modélisées (français, anglais). La méthode d'identification de la langue est simple : elle utilise des antidictionnaires (ou stop listes) propres à chaque langue. Il s'agit de compter, pour chaque message, le nombre de mots outils (articles, prépositions…). Nous avons pu recenser plus de 355 termes en langue française et 571 en langue anglaise que nous avons regroupés dans deux *stoplist*. Ces listes triées et indexées sont enregistrées dans une base de données qui sera chargée en mémoire lors du démarrage du logiciel sous forme de deux vecteurs triés pour optimiser le temps de recherche.

Cette méthode permet aussi de signaler les messages bilingues et ceux qui ne sont pas dans l'une de ces langues. Par ailleurs, le système est incrémental et permet facilement la prise en compte de nouvelles langues (ajouter un antidictionnaire

propre à chaque nouvelle langue). Puis, ayant connaissance de la langue, un ensemble de règles de lemmatisation est appliqué sur les différents mots du message pour réduire les variantes morphologiques à une forme commune (mettre les verbes à l'infinitif, supprimer les formes plurielles…). A cet effet, nous avons utilisé un analyseur morphologique flexionnel, analyseur FLEMM (Namer, 2000), fourni par l'ATILF (Analyse et Traitement Informatique de la Langue Française).

L'analyseur FLEMM ainsi que l'extraction de certaines propriétés (d'ordre syntaxique) nécessitent une phase d'étiquetage préalable. En effet, nous avons utilisé un étiqueteur morphosyntaxique, analyseur de Brill (1992). La figure 2 présente un exemple de résultat de l'étiqueteur.

Message : *Bonjour Omar. Tout s'est très bien passé. C'est dommage que tu n'aies pas pu venir, car c'est le moment d'apprendre beaucoup de choses. C'est le moment aussi pour prendre des contacts. Bon courage. Alain.*

Message étiqueté : *Bonjour/INJ Omar./SBP:sg Tout/DTN:sg s'/PRV:sg est/ECJ:sg très/ADV bien/ADV passé/ADJ2PAR:sg ./. C'/PRV:sg est/ECJ:sg dommage/SBC:sg que/SUB$ tu/PRV:sg n'/ADV aies/ACJ:sg pas/ADV pu/VPAR:sg venir/VNCFF ,/, car/COO c'/PRV:sg est/ECJ:sg le/DTN:sg moment/SBC:sg d'/PREP apprendre/VNCFF beaucoup/ADV de/PREP choses/SBC:pl ./. C'/PRV:sg est/ECJ:sg le/DTN:sg moment/SBC:sg aussi/ADV pour/PREP prendre/VNCFF des/DTC:pl contacts/SBC:pl ./. Bon/ADJ:sg courage/SBC:sg ./. Alain/SBP:sg ./.*

Jeu d'étiquettes : INJ : *Interjection, Onomatopée...* ***SBP :*** *Substantif, nom propre ou à majuscule.* ***DTN :*** *Déterminant.* ***PRV :*** *Pronom « supporté » par le verbe (conjoint, clitique).* ***ECJ :*** *Verbe « être », conjugué.* ***ADV :*** *Adverbe.* ***ADJ2PAR :*** *Participe passé adjectival (non après auxiliaire).* ***SBC :*** *Substantif, nom commun.* ***SUB$:*** *Subordonnant possible. = Code par défaut de « que ».* ***ACJ :*** *Verbe « avoir », conjugué.* ***VPAR :*** *autre Verbe, non conjugué, participe passé après « avoir ».* ***VNCFF :*** *autre Verbe, non conjugué, infinitif.* ***COO:*** *Coordination.* ***PREP :*** *Préposition.* ***DTC :*** *Déterminant de groupe nominal, contracté.* ***Sg:*** *Singulier…*

Figure 2. *Résultat de l'étiqueteur Brill*

3.2. *Analyseur linguistique*

A l'issue de l'étape de prétraitement, le message est donné à un analyseur linguistique qui a pour but d'identifier et d'extraire les différentes propriétés linguistiques permettant de caractériser le contenu de chaque message. Il est indépendant de tout domaine d'application : il reçoit en entrée un texte, il délivre en sortie la représentation associée.

En effet, l'analyseur fait passer le message par les différents modèles linguistiques réduits de base et construit en sortie le vecteur message associé. L'objectif de chaque niveau est d'analyser un message et d'en extraire un ensemble de caractéristiques.

En sortie, un message est représenté par un ensemble d'entités lexicales et par une suite de caractéristiques (énonciatives, structurelles, syntaxiques et d'architecture textuelle). Les caractéristiques sont représentées par des variables dans le vecteur message. Ces variables dénotent la présence ou l'absence de ces caractéristiques dans le message.

Le message est représenté conceptuellement par un espace vectoriel de *k* dimensions :

$$M = \{(T1,W1), (T2,W2...(Tk,Wk)\}$$

Ti représente la ième caractéristique, *Wi* le poids et *k l'*espace des caractéristiques. Cette représentation constitue l'entrée du module de classification et de filtrage. Elle est donc créée dynamiquement à chaque récupération d'un nouveau message. Ce vecteur sera propagé à travers les différentes couches du réseau de neurones pour donner en sortie le type du message et prendre en compte des actions de filtrage telles que supprimer, sauvegarder, signaler… définies par l'utilisateur.

3.3. *Module de classification et de filtrage*

Ce module est constitué de deux parties : une partie classification et une partie filtrage. La classification permet d'affecter à un message une catégorie constituant en quelque sorte un préfiltrage (figure 3). En effet, une idée pour classer les messages est de créer des espaces de messages (un espace pour chaque type). Chaque nouveau message se trouvant proche des messages de l'un des espaces définis est alors considéré comme pertinent pour cet espace. Donc, pour mieux classer un message, le module utilise un modèle de connaissances qui représente et modélise une typologie de messages, dont la connaissance est construite initialement sur la base d'analyse de traits linguistiques associés à chaque espace de messages.

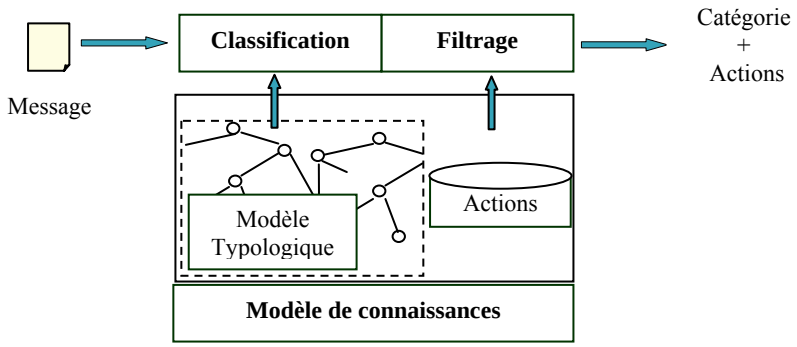

Figure 3. *Processus de classification et de filtrage*

Après l'étape d'analyse linguistique, le message passe par un arbre de classification qui lui affecte automatiquement une catégorie. Ensuite, il passe par un processus de filtrage qui permet, selon les spécificités de l'utilisateur, d'entreprendre des actions de filtrage. Ce processus est étroitement lié et adapté aux souhaits de l'utilisateur.

4. Modèles linguistiques réduits

Les connaissances linguistiques (données lexicales, structure du texte…) sont de plus en plus utilisées dans les systèmes d'analyse de textes, par exemple, pour identifier l'information pertinente (Poibeau, 1999 ; Marcu, 1997 ; Minel *et al.,* 2001).

Nous avons défini et identifié un ensemble de propriétés, dont nous avons automatisé la recherche, qui sert à caractériser les textes. Il s'agit d'un ensemble d'indicateurs sur le texte qui permet de le situer par rapport aux autres, de rapprocher les textes qui appartiennent à la même classe ou éloigner ceux qui appartiennent à des classes différentes.

Pour notre application de filtrage de courriels, l'étude statistique de notre corpus nous a permis d'ajouter d'autres indices supplémentaires qui sont spécifiques à la messagerie électronique pour tenter d'améliorer les performances du système.

4.1. *Modèle lexical*

Il représente l'ensemble des entités lexicales les plus pertinentes du domaine traité. Il est généré automatiquement à partir de corpus. Il constitue le noyau de base sur lequel repose toute méthode d'identification et de représentation des documents textuels. Pour notre corpus, nous avons défini et identifié, d'une façon automatique, deux types d'entités : mots simples et mots composés.

– **Mots simples (MS) :** représentant le vocabulaire de base. Il est constitué d'unités linguistiques spécifiques les plus pertinentes (mot, lemme…).

Initialement, chaque message du corpus subit un prétraitement qui permet d'éliminer les mots outils (articles, prépositions…). Ensuite, nous réduisons les variantes morphologiques à une forme commune (souvent appelé terme).

Le vocabulaire construit lors du traitement de notre corpus de courriers comprend initialement 54 760 mots. Le critère utilisé pour le réduire est la mesure de l'information mutuelle (Yang et Pedersen, 1997).

L'information mutuelle MI (*t, C*) mesure la dépendance d'un terme *t* et d'une classe *C.* Elle est définie par :

$$MI(t,C) \approx \frac{N * p}{(p + p')(p + q)}$$

avec :

p : est le nombre de messages de classe *C* qui contiennent le terme *t ;*

q : est le nombre de messages qui ne sont pas de classe *C* mais qui contiennent le terme *t ;*

p' : est le nombre de messages de classe *C* qui ne contiennent pas le terme *t* ;

q' : est le nombre de messages qui ne sont pas de classe *C* et qui ne contiennent pas le terme *t ;*

N : est le nombre total de messages du corpus.

Cette mesure numérique permet de déceler les mots qui « s'attirent », c'est-à-dire qui tendent à apparaître en même temps. Une information mutuelle élevée entre un terme et une classe est le signe d'un lien fort entre ces deux éléments. Le vocabulaire est réduit alors à 600 termes dont les informations mutuelles sont les plus élevées. Voici un extrait du vocabulaire sélectionné pour les domaines considérés (tableau 1).

Spam
business, time, money, free, price, product, credit, order, opportunity, guarantee, click, marketing, investment, risk, advertisement, sex, travel, miracle…
Personnel
a+ , absence, actuellement, besoin, beaucoup, bises, bisous, bonheur, bonjour, bonne, contacter, courage, courrier, dérangement, désolé, dieu, dommage, embrasser, espérer, essayer, excuser, famille, galère, heureuse, job, joie, maman, merci, nouvelles, ok, papa, plaisir, salut, samedi, super, vacances, visa, visite, vœux, voiture, voix, voyage…
Professionnel
actes, appel, calendrier, cher, collègue, comité, communication, conférence, cordial, critères, date, langue, madame, monsieur, plaisir, salutation, soumission…

Tableau 1. *Vocabulaire de base*

– Mots composés ou phrases très courtes (MC) : vocabulaire généré à partir des listes *bigrammes* et *trigrammes* apprises par le système. Voici un extrait (tableau 2).

Spam
bulk email, business opportunity, credit card, credit repair, financial news, free erotic, free investment, half price, home business, home worker, immediate release, investment report, limited time, live sex, low price, major credit, money order, offer valid, order by phone, order report, phone number, return address, sex stories, special bonus, take action, time offer, xxx video…
Personnel
à bientôt, à plus, à toute, après-midi, as-tu, aurais-je, deviens-tu, dis-moi, es-tu, fais-tu, grosso-modo, mi-temps, parce que, peux-tu, puis-je, rendez-vous, sais-tu, week-end…
Professionnel
appel a communication, cher collègue, comite de lecture, comite de programme, comite d'organisation, critères de sélection, date limite de soumission, journées d'étude, salutations distinguées, final camera, ready copy, method of submission, notification of acceptance, notification of workshops, notification to authors, organized by, organizing committee, paper submission, paper submission form, selection criteria, submitted papers…

Tableau 2. *Vocabulaire composé*

4.2. *Modèle concernant la mise en forme matérielle (l'architecture du texte)*

Ce modèle traite de la mise en relation entre une mise en forme matérielle d'un texte et sa structure logique. Il permet de localiser et d'identifier la nature des zones textuelles (entête, titre, corps, paragraphe, section, listes, tableaux…). La ponctuation des textes est une extension des signes de ponctuation de la phrase (elle peut inclure le format de titres, la forme des paragraphes, les notes de bas de page). Cette ponctuation textuelle permet de percevoir des objets textuels (des chapitres, des sections, des paragraphes…) et aussi des relations entre ces objets textuels (inclusions, liens sémantiques…). L'ensemble des objets et des relations définit, selon Virbel, l'architecture du texte (Virbel et Luc, 2001). La syntaxe de la ponctuation des textes repose sur un ordre de marques lexicales en fonction de leur contenu (comme introduction et conclusion) et sur des propriétés typo-positionnelles (par exemple, ce qui est centré en tête de page est un titre).

Dans le cadre de ce travail, nous ne cherchons pas à retrouver précisément chaque expression sémantique qui nous permettrait de mettre à jour chaque acte de langage qui se trouverait dans toutes les configurations particulières ; cependant, nous utilisons les procédés qu'utilise la syntaxe de la ponctuation de texte pour

alimenter notre système de critères distinctifs entre les courriers. Voici quelques exemples d'indicateurs :

– titres, sections, paragraphes ;

– introduction, conclusion ;

– images, dessins, ponctuation (?, !…) ;

– p.s. (à la fin), fichiers attachés ;

– nombre de destinataires (champ *to*), le domaine des adresses émettrices ;

– type du message : text/html, la longueur de l'entête des messages ;

– longueur moyenne des messages, longueur moyenne des phrases ;

– mots en majuscule, abréviations, caractères non alphanumériques ($, #…) ;

– caractères numériques ;

– horaire d'envoi (nuit/jour).

4.3. *Modèle énonciatif*

Lorsqu'on aborde le sens des unités linguistiques, on est amené à les relier à leur référence telle que l'énonciateur : c'est-à-dire qu'il faut porter le regard sur l'acte par lequel le discours est produit (Benveniste, 1966). Ce modèle concerne donc le locuteur ou l'énonciateur : nous cherchons à identifier un ensemble d'indices linguistiques qui font référence à l'énonciateur tels que les pronoms personnels, les formes verbales, les formes temporelles… pour alimenter notre système de filtrage. Voici quelques exemples d'indicateurs :

– 1re pers du singulier, 2e pers du singulier, 1re pers du pluriel, 2e pers du pluriel, 3e pers (singulier, pluriel) ;

– déterminants (mon, ton, son, ce…) ;

– discours rapporté direct ;

– retours de courriers (réponses) : présence de (Re :) ;

– énoncer : admettre, dire, déclarer, remarquer, protester… ;

– penser, croire, révéler, supposer, estimer…

4.4. *Modèle structurel*

Des théories basées sur l'étude du discours attribuent une représentation arborescente à la structure du texte. Les feuilles de l'arbre représentent des énoncés linguistiques ou leurs représentations sémantiques. La structure de l'arbre est basée sur la nature de la relation qui est établie entre les empans de textes que l'on relie entre eux. Cependant, ces relations qui existent entre chaque proposition sont dans la plupart des cas implicites. Par exemple, une relation causale entre deux énoncés peut

être établie sur une connaissance du monde partagée par les deux locuteurs. De ce fait sans une analyse sémantique « profonde » du texte nous n'avons accès qu'aux termes explicites de ces relations. En effet, un certain nombre de marqueurs linguistiques (un ensemble de mots-clés) précisent la relation ou un ensemble de relations potentielles entre les deux segments de textes reliés par ce marqueur.

Dans un premier temps les seuls indices que nous avons sur la structure des textes sont les cas d'explicitation sous forme de *mots-clés* des relations « rhétoriques » des textes. La collecte des mots-clés se fait sur la base de travaux divers sur des relations (Marcu, 1997).

Nous avons instauré une hiérarchie des termes basée sur leur position. Marcu remarque que la position de certains mots-clés est liée à leur fonction qui est soit discursive, soit syntaxique. Nous avons distingué trois types de positions en début de paragraphe, en début de phrase, après une virgule et les autres positions. Pour chacune de ces positions nous avons pondéré la relation différemment en fonction de l'importance des empans de texte qui peuvent être mis en relation. Un mot-clé en début de paragraphe peut mettre en relation deux paragraphes, alors qu'un mot-clé en début de phrase peut mettre en relation deux phrases : nous accordons un poids plus important au mot-clé qui porte sur l'empan le plus large donc à celui qui se trouve entre deux paragraphes. Voici quelques exemples d'indicateurs :

- addition (à cela s'ajoute qu, ainsi qu, aussi, d'autre part, de plus…) ;
- analogie (c'est-à-dire, comme, de la même façon, de même...) ;
- but (pour qu, de sorte qu...) ;
- cause (afin qu, c'est pourquoi...) ;
- exemple (à savoir, par exemple...) ;
- focus (particulièrement, précisément...) ;
- intensité (assez, au point qu...).

4.5. *Modèle syntaxique*

L'analyse syntaxique est une composante très importante dans le traitement automatique des langues. L'analyse syntaxique regroupe divers courants qui diffèrent sur les objectifs visés et sur les méthodes employées. Les méthodes couvrent par exemple, les approches stochastiques, les approches d'analyse locale et les approches plus traditionnelles d'analyse complète (ou profonde). Les objectifs vont de la segmentation en syntagmes à l'analyse profonde avec une grammaire à large couverture, en passant par des analyseurs robustes et/ou superficiels. Néanmoins, cette diversité dans les méthodes et objectifs reflète une certaine complémentarité plutôt qu'une opposition absolue.

Dans le cadre de ce travail, nous ne cherchons pas à retrouver précisément la structure syntaxique de chaque énoncé, nous cherchons à identifier un ensemble

d'indices syntaxiques pour alimenter notre système de filtrage de courriels. Voici quelques exemples d'indicateurs :

– taux de pronoms ;

– taux de déterminants ;

– taux de noms propres;

– nominalisations ;

– nombre d'adverbes (temps/lieu) ;

– nombre d'adjectifs ;

– infinitifs, participes passés ;

– coordinations, négations, démonstratifs, indéfinis (anaphoriques) ;

– relatifs (sujet/objet) ;

– subordinations, interrogations ;

– interjections, abréviations ;

– formes actives/formes passives.

L'extraction de ces indices d'ordre syntaxique nécessite une phase d'étiquetage préalable. En effet, l'analyse consiste à rechercher dans le texte, étiqueté par l'analyseur Brill, des séquences d'étiquettes correspondantes à chaque type d'indices.

5. Profils de l'utilisateur

La modélisation des intérêts de l'utilisateur est une tâche importante pour un système de filtrage de l'information. L'efficacité du filtrage est étroitement liée à cette modélisation. La mise en pratique d'un modèle utilisateur est difficile car l'utilisateur lui-même a des difficultés à décrire ses attentes de manière formelle et explicite. Pour des raisons ergonomiques, une solution simple est de présenter à l'utilisateur des types ou classes de messages clairement identifiables plutôt qu'un ensemble de propriétés linguistiques complexes et inexploitables. L'utilisateur pourra alors créer ses propres profils (classes) :

– soit utiliser ou combiner des types prédéfinis ;

– soit proposer de nouveaux types introduits dans le système sous formes différentes :

- **sous forme de mots-clés** : l'utilisateur introduit une liste de mots-clés et pour chacun, il associe un poids qui représente son degré d'importance. Ce type de profils est amélioré et augmenté de propriétés linguistiques par un apprentissage. Le système se chargera de collecter et de construire le corpus d'apprentissage ;

- s**ous forme de texte** : ici, l'utilisateur introduit des textes et le système extraira des mots-clés et des propriétés linguistiques et leur attribuera un poids ;

- **sous forme d'url** (ex. : adresse d'un serveur).

Chaque forme subira un traitement spécifique pour être représentée dans le système.

5.1. *Profil de base*

L'existence d'une typologie de messages et d'un corpus de référence faisant défaut, nous nous sommes donc limités à trois types génériques de messages bien particuliers pour construire le modèle de connaissances ou utilisateur (profil de base) : les messages *personnels*, *professionnels* et les messages indésirables (appelés *Spam*).

Les messages personnels regroupent tous les messages familiaux, ceux provenant d'amis, ainsi que les messages personnels-professionnels (collègue-collègue, étudiant-professeur...).

Les messages *professionnels* regroupent les appels à communication, les annonces de livres, les articles, les messages de directions, d'institutions...

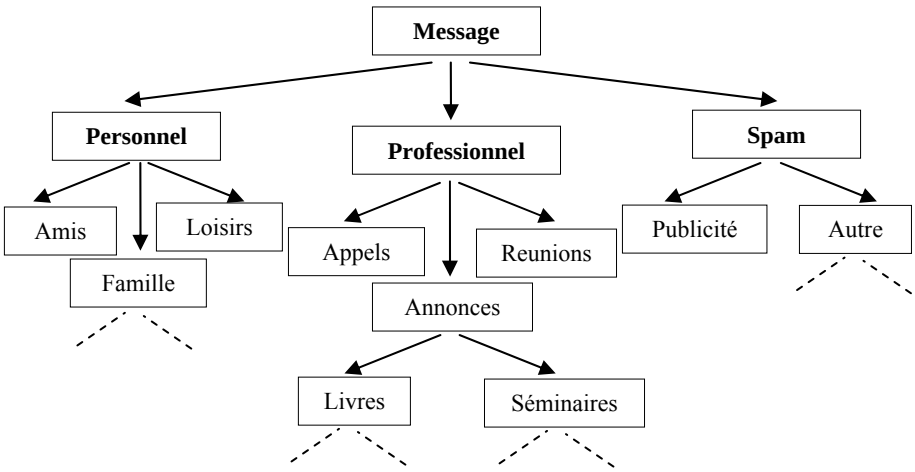

Figure 4. *Typologie de messages*

Enfin, les messages non sollicités et indésirables appelés *Spam* qui polluent nos boîtes emails. Ils constituent le centre de contre-intérêt de l'utilisateur. Il s'agit de messages publicitaires proposant des services, des produits miraculeux (maigrir en un temps record...), offres de voyages à prix attractif, opportunités d'investissement pour devenir riche en peu de temps, propositions de cartes de crédit à taux d'intérêt réduit, messages pornographiques... Le *spam* est un phénomène mondial et massif. Il cause de multiples désagréments tels que l'engorgement des boites emails et des

serveurs emails, dilution des messages utiles, perte de temps et d'espace... Devant l'importance de ce phénomène, il est donc nécessaire aujourd'hui, afin d'aider l'utilisateur submergé de mails, d'élaborer des outils efficaces capables de traiter et de filtrer le courrier électronique, et plus particulièrement le courrier *spam*.

Cette typologie constitue donc une sorte de profil de base qui permettra d'aider l'utilisateur à décrire et élargir ses propres profils (le modèle utilisateur). Pour déterminer les propriétés linguistiques utiles pour notre modèle de base, nous faisons passer chaque message du corpus par les différents modèles linguistiques réduits de base. En sortie de cet apprentissage, nous obtenons une typologie constituée seulement de propriétés linguistiques utiles (poids supérieur à un certain seuil) pour chaque type considéré.

5.2. *Modèle adopté*

La typologie générique des messages est réalisée (modélisée) par un réseau de neurones non récurrent (absence de boucles) à trois couches (figure 5). Une couche en entrée qui reçoit les entrées du réseau. Une couche cachée représentant l'ensemble des connaissances (profils). Une couche de sortie qui représente les types de messages (*spam, personnel* et *professionnel*).

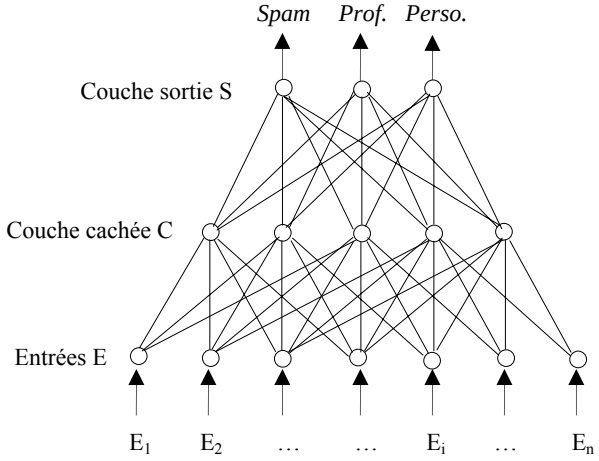

Figure 5. *Architecture d'un réseau à trois couches*

L'avantage des réseaux de neurones est leur adaptation aux applications qui traitent des données bruitées et dont la solution est inconnue ou très difficile à formaliser (Dreyfus *et al.*, 2002).

6. Apprentissage et correction

Nous appelons apprentissage la procédure qui consiste à estimer les paramètres d'un système afin que celui-ci remplisse au mieux la tâche qui lui est affectée. Dans un réseau de neurones, la connaissance est codée par la valeur des poids des différentes connexions. Ce codage est estimé par apprentissage. Nous avons travaillé avec un corpus de 800 messages dit « base d'apprentissage » composé de messages *spam*, *personnel* et *professionnel*. Il est annoté manuellement.

Le réseau est entraîné sur cette base d'apprentissage dans le but de catégoriser correctement un nouveau message par l'algorithme de propagation arrière ou rétro-propagation qui consiste à corriger les poids des connexions en fonction des erreurs commises. La correction se fait de la couche de sortie à la couche d'entrée. L'apprentissage utilisé est dit supervisé, c'est-à-dire que nous testons le réseau dans des situations connues et nous cherchons à obtenir la sortie voulue. Nous effectuons alors la modification des poids pour retrouver cette sortie imposée.

6.1. *Schéma d'algorithme*

L'algorithme d'apprentissage est décrit comme suit :

– étiqueter manuellement chaque message du corpus (*spam*, *personnel* ou *professionnel*) ;

– faire passer le corpus par les différents modules d'analyse pour extraire les différentes propriétés linguistiques et avoir la représentation vectorielle associée de chaque message ;

– initialiser les paramètres du réseau : les poids, les seuils, le *pas d'apprentissage* et le nombre d'itérations. Les poids doivent être initialisés à des petites valeurs aléatoires entre *– 0,5* et *+ 0,5* (Dreyfus, 2002). Au départ, les poids des connexions entre neurones des différentes couches sont définis par défaut à *0,5*. Le choix du *pas d'apprentissage* est très important car s'il est trop petit, la convergence du réseau risque d'être très lente ; s'il est trop grand, il y a risque d'oscillation. Généralement, le *pas* doit être compris entre *0,05* et *0,25* (Dreyfus, 2002). Nous avons initialisé le *pas* à *0,1* ;

– lancer l'apprentissage qui consiste à :

- calculer la sortie du réseau pour chaque message

$$S(T_j) = f(E(T_j))$$

$$E(T_j) = \sum_j S(C_j) * P_{js}$$

$$S(C_j) = f(E(C_j))$$

$$E(C_j) = \sum_i S(t_i) * P_{ij}$$

où :

S(T_j) : la valeur du neurone de sortie d'indice *j ;*

E(T_j) : les valeurs des entrées du neurone de sortie d'indice *j ;*

S(C_j) : la sortie du neurone caché d'indice *j ;*

E(C_j) : les entrées du neurone caché d'indice *j ;*

S(t_i) : la sortie du neurone en entrée d'indice *i ;*

P_{ij} : la valeur du poids de la connexion du neurone d'indice *i* de la couche d'entrée vers le neurone d'indice *j* de la couche cachée ;

P_{js} : la valeur du poids de la connexion du neurone d'indice *j* de la couche cachée vers le neurone d'indice *s* de la couche de sortie ;

f : la fonction sigmoïde, définie par (Davalo & Naim, 1983) : $f(x) = \frac{1}{1+e^{-x}}$;

- comparer et calculer l'erreur :

$$DS = S*(1 - S)*(1 - S)$$

$$DC = C_j *(1 - C_j)*(P_{js} *DS)$$

où :

DS : erreur du réseau pour la couche de sortie ;

DC : erreur du réseau pour la couche cachée ;

- mettre à jour les paramètres du réseau par rétropropagation (de la couche sortie vers la couche entrée) : ajuster les poids ;

$$P_{ij}(t + 1) = P_{ij}(t) + r*DC*S(t_i)$$

$$P_{js}(t + 1) = P_{js}(t) + r*DS*S(C_j)$$

où :

r : le taux d'apprentissage ;

t : le numéro du cycle ;

– test d'arrêt : la convergence de l'algorithme de *rétropropagation* est assurée par un test consistant à fixer le nombre d'itérations ou bien à arrêter l'apprentissage dès que l'erreur devient inférieure à un certain seuil. Si ce seuil est très proche de *0*, il y a un grand risque de *surapprentissage* ; au lieu de produire une bonne généralisation, le réseau se concentre sur les particularités des exemples d'apprentissage. En pratique, le test d'arrêt est lié aux mesures des performances du réseau. Pour mesurer les performances du réseau, il convient de constituer, outre l'ensemble d'apprentissage utilisé pour déterminer les poids, un ensemble de tests constitué d'exemples différents de ceux de l'ensemble d'apprentissage à partir

duquel nous estimons les performances du réseau après un apprentissage. Nous alternons des étapes d'apprentissage sur l'ensemble d'apprentissage et de mesure des performances sur l'ensemble de tests jusqu'à atteindre des résultats satisfaisants. En effet, l'apprentissage consiste donc à trouver l'ensemble des paramètres *w* du réseau qui rendent la fonction de coût des moindres carrés *J(w)* minimum définie par la formule suivante (Dreyfus, 2002) :

$$J(w) = \frac{1}{2} \sum_{k=1}^{N_a} [Y_p(x_k) - g(x_k, w)]^2$$

où :

N_a *:* nombre d'exemples de l'échantillon apprentissage ;

x_k *:* vecteur des valeurs des variables pour l'exemple k ;

w : vecteur des poids du réseau ;

$g(x_k, w)$ *:* valeur calculée par le réseau ;

$Y_p(x_k)$ *:* valeur de la mesure correspondante ;

Il s'agit d'une technique itérative qui modifie les paramètres *w* du réseau jusqu'à ce que *J(w)* soit minimum. Ensuite, mesurer l'indice de performance qui représente l'erreur quadratique moyenne commise sur l'ensemble de tests désignée par *EQMT* (Dreyfus, 2002) :

$$\text{EQMT} = \sqrt{\frac{1}{N_t} \sum_{k=1}^{N_t} [Y_p(x_k) - g(x_k, w)]^2}$$

où N_t est le nombre d'exemples de l'ensemble test.

Enfin, comparer *EQMT* à l'erreur quadratique moyenne commise sur l'ensemble d'apprentissage *EQMA* :

$$\text{EQMA} = \sqrt{\frac{1}{Na} \sum_{k=1}^{Na} [Y_p(x_k) - g(x_k, w)]^2}$$

où N_a est le nombre d'exemples de l'ensemble apprentissage.

6.2. *Correction*

Le facteur intelligent de notre système est sa faculté d'apprendre et d'améliorer l'efficacité du filtrage. Une fois un message reclassé, le système doit réorganiser les messages en fonction des courriels lus par l'utilisateur. Le système dispose d'un

apprentissage assisté appelé ***feed-back*** où l'utilisateur peut soit donner un avis direct sur le message lu, soit déplacer un message d'une classe à une autre ou créer une nouvelle classe (cas de mauvaise classification par le système), ce qui lui permet d'approcher la pertinence de l'utilisateur et de s'adapter ainsi à ses besoins. L'apprentissage agit sur le profil qui consiste à modifier les poids dans le but d'améliorer la réponse du système. L'utilisateur peut aussi ajouter et supprimer des mots et des profils à sa demande. Le modèle de filtrage est donc recalculé à partir de cette nouvelle base de messages construite au fur et à mesure de l'utilisation du système. Cependant, la qualité du modèle dépend énormément de la qualité et de la taille du corpus utilisé. Par conséquent, cette opération nécessite, généralement, plusieurs sessions d'utilisation du système.

7. Evaluation

Nous avons mené des tests pour 1) mesurer l'importance et le rôle de l'information linguistique dans la représentation des messages, 2) mesurer les performances du système de classification du point de vue précision et rappel, 3) et montrer comment l'opération d'apprentissage agit sur l'efficacité du filtrage.

7.1. *Corpus*

Pour effectuer nos tests, nous avons travaillé avec un corpus de 1 200 messages construit à partir d'un ensemble de messages que nous avons collectés pendant quatre mois. Il regroupe une variété de types de messages. Nous nous sommes limités à trois types génériques de messages pour construire le modèle linguistique (profil de base) : 700 courriers de classe *spam* et 500 non *spam* (35 % des messages sont de type personnel et 65 % de type professionnel). Nous avons divisé le corpus en une base d'apprentissage et une base de tests (tableau 3).

Catégorie	Base d'apprentissage	Base de test
Spam	470	230
Personnel	135	65
Professionnel	200	100

Tableau 3. *Découpage du corpus de travail*

7.2. C*ritères d'évaluation*

Pour mesurer les performances, nous utilisons les mesures de *précision* et de *rappel*. Nous déterminons également la performance globale du système en calculant le pourcentage d'erreur et de succès.

Jugement du Système ⇓	Jugement de l'utilisateur	
	C	¬*C*
C	α	β
¬*C*	γ	δ

Tableau 4. *Critères d'évaluation*

avec :

α : messages de classe *C* correctement filtrés (classés) par le système ;

β : messages n'appartenant pas à la classe *C* incorrectement filtrés par le système ;

γ : messages de classe *C* incorrectement non filtrés (rejetés) par le système ;

δ : messages n'appartenant pas à la classe *C* correctement non filtrés par le système.

Les mesures *rappel* et *précision* pour le filtrage de messages de classe *C* sont :

$$\text{rappel} = \frac{\alpha}{\alpha + \gamma} \qquad \text{précision} = \frac{\alpha}{\alpha + \beta}$$

Les mesures globales *erreur* et *précision* du système sont *:*

$$\textit{erreur\_globale} = \frac{\beta + \gamma}{\alpha + \beta + \gamma + \delta}$$ c'est le rapport entre le nombre total de messages incorrectement filtrés et incorrectement non filtrés et le nombre total de messages de la base de test ;

$$\textit{précision\_globale} = \frac{\alpha + \delta}{\alpha + \beta + \gamma + \delta}$$ c'est le rapport entre le nombre total de messages correctement filtrés et correctement non filtrés et le nombre total de messages de la base de test.

7.3. *Expériences*

Nous présentons, dans ce qui suit, les performances de notre système de filtrage dans plusieurs cas de configuration.

7.3.1. *Expérience 1 : performances en fonction des caractéristiques lexicales*

Nous mesurons les performances du système en considérant tout d'abord un modèle de base constitué uniquement de mots simples. Les connaissances du système sont décrites par trois profils et introduites dans le système sous deux formes différentes : modélisation manuelle et modélisation automatique.

– Modélisation manuelle : nous avons considéré pour effectuer cette modélisation trois utilisateurs (connaissant bien le domaine considéré). Chaque utilisateur a défini, pour chaque type de profil considéré, une liste de mots-clés. Le vocabulaire de base utilisé par le système pour représenter les textes du corpus est l'union des listes des trois utilisateurs. La liste de mots-clés définie par l'ensemble des utilisateurs est résumée dans le tableau 5.

Spam
argent, banque, fille, film, football, marketing, médicament, sexe, sport, téléphoner par Internet, travail à distance, virus, DVD, voyage...
Personnel
bisous, bonjour, embrasser, famille, frère, Mohamed, maman, Omar, papa, Sarah, salut, sœur, vacances...
Professionnel
appel a communication, comité de lecture, date limite de soumission, journées d'étude, conférence, langue...

Tableau 5. *Mots-clés introduits par l'utilisateur*

– Modélisation automatique : le système se charge de la modélisation en se basant sur un apprentissage à partir du corpus.

Les résultats obtenus avec une modélisation automatique du profil sont nettement meilleurs qu'avec une modélisation manuelle (ce qui montre la difficulté de l'utilisateur à décrire ses propres profils). De plus, nous constatons que certains mots corrèlent avec certains types de messages considérés mais statistiquement sont insignifiants (valeur faible). Ce qui nous a poussé à modifier l'importance des différents mots tout en gardant un traitement générique sans l'intervention de l'utilisateur. Le système attribue une forte valeur du poids aux mots qui sont uniques

dans chaque catégorie par rapport à ceux qui se trouvent dans plusieurs catégories. Les résultats des tests étaient meilleurs (tableau 6).

Catégories	**Modélisation manuelle**	**Modélisation automatique**	
		Avant modification	**Après modification**
Personnel	45 %	83 %	85 %
Professionnel	72 %	88 %	91 %
Spam	67 %	87,7 %	90 %

Tableau 6. *Performances en fonction des caractéristiques lexicales*

7.3.2. *Expérience 2 : performances en fonction des mots composés*

Au début de l'expérience, nous ajoutons un ensemble de mots composés (MC) au modèle de base constitué initialement de mots simples (MB).

Caractéristiques	**Performance globale**					
	Personnel		*Professionnel*		*Spam*	
	Erreur Globale	Précision Globale	Erreur Globale	Précision Globale	Erreur Globale	Précision Globale
MB	17 %	83 %	12 %	88 %	12,3 %	87,7 %
MB + MC	17 %	83 %	11 %	89 %	13 %	87 %
MB + MC + Pondération	17 %	83 %	9 %	91 %	8,6 %	91,4 %

Tableau 7. *Performances en fonction des mots composés*

Nous ne constatons pas une amélioration des performances. En effet, les mots composés corrèlent avec les types de messages considérés mais statistiquement sont insignifiants (valeur faible). Ensuite, nous avons modifié l'importance de ces différents mots composés en leur attribuant une forte valeur du poids. Les résultats des tests étaient nettement meilleurs (ex. : 91 % pour le profil *spam*).

7.3.3. *Expérience 3 : performances en fonction des caractéristiques linguistiques*

Dans un premier temps, nous avons considéré toutes les propriétés linguistiques sans restriction pour la construction du modèle. Les résultats sont résumés dans le tableau 8.

Catégories	Performances
Personnel	74 %
Professionnel	79 %
Spam	82 %

Tableau 8. *Les performances du modèle sans restriction*

Nous constatons que les performances du système sont dégradées. En effet, certaines caractéristiques ne corrèlent pas avec certains types de messages (valeur = 0). Nous avons donc testé les performances lorsque nous réduisons le nombre de propriétés du vocabulaire en imposant un seuil sur les occurrences des propriétés à considérer pour les différentes représentations des courriers. Les résultats sont donnés dans le tableau 9.

Catégories	Modèle de base	Modèle de base + PL
Personnel	85 %	92 %
Professionnel	91 %	93 %
Spam	90 %	95 %

Tableau 9. *Les performances avec restriction*

Nous constatons que les performances globales du système sont améliorées. Ceci s'explique par le fait que les messages rejetés par le système (1re expérience) par absence de mots-clés ou valeur très faible sont acceptés cette fois ci, et ceci à cause de la présence de certaines propriétés linguistiques (PL). Par exemple, les messages personnels sont caractérisés par l'utilisation de pronoms personnels (1re et 2e personne).

7.3.4. *Expérience 4 : mesurer l'importance et le rôle de l'apprentissage assisté*

Dans cette expérience, l'utilisateur a la possibilité de créer ses propres profils : soit utiliser ou combiner des types prédéfinis, soit proposer de nouveaux types. Nous considérons un utilisateur avec trois profils différents :

– un profil *P1* choisi parmi les 3 proposés par le système,

– un nouveau *P2* créé en proposant une liste de mots clés,

– et enfin un autre profil *P3* choisi et modifié par l'utilisateur.

Nous constatons que les résultats varient d'un profil à l'autre. Après un certain temps d'apprentissage, les résultats des profils *P1* et *P3* restent presque stables, l'apprentissage n'améliore pas vraiment les résultats (profils satisfaisants). Par

contre, dans le cas du profil *P2*, le modèle converge vers un modèle de filtrage satisfaisant, mais lentement. En effet, le modèle nécessite plusieurs sessions d'apprentissage assisté pour améliorer la qualité de ses résultats. Il est donc nécessaire de lancer l'apprentissage *feedback* régulièrement, par exemple, après chaque session de filtrage.

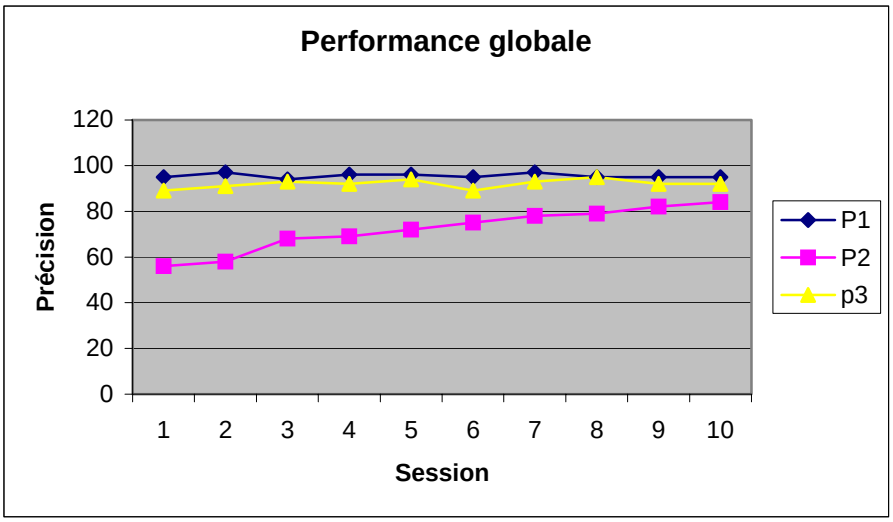

Figure 6. *Apprentissage assisté*

8. Discussion

A travers les différentes expériences réalisées, nous avons montré l'applicabilité et l'adaptabilité d'une approche linguistique au processus de filtrage. En effet, après les tout premiers tests, nous avons remarqué que les résultats semblaient plutôt satisfaisants. Mais nous ne pouvons pas affirmer que certaines propriétés telles que les propriétés concernant la structure matérielle constituent une connaissance suffisante de discrimination de messages. Néanmoins, la probabilité d'avoir un mail d'un certain type est plus forte quand ces caractéristiques sont vérifiées.

Les résultats obtenus sur notre corpus se révèlent constructifs. Néanmoins, il serait intéressant d'extrapoler l'étude sur d'autres types de textes pour étendre la liste des critères et tester l'adaptabilité de l'approche.Les expériences menées sur notre corpus de messages, très modeste, nous ont permis de valider :

– le recours aux connaissances linguistiques, sous forme de modèles linguistiques réduits, pour améliorer les performances d'un système de filtrage d'information. Ces connaissances, portant sur la structure et le contenu, sont classées en plusieurs niveaux linguistiques ;

– l'approche évolutive par apprentissage automatique, passage obligé dans la conception et l'amélioration des performances d'un système de filtrage d'information dynamique ;

– la portabilité du système : les connaissances de base sont indépendantes du domaine d'application (les modèles linguistiques réduits). Les connaissances spécifiques à l'application (email) sont générées automatiquement. En effet, le profil de l'utilisateur est calculé par analyse automatique du contenu qui permet de produire un ensemble de termes et de propriétés linguistiques le caractérisant. De plus, notre système est complètement indépendant du domaine de connaissances. Il a une structure modulaire, lui permettant éventuellement de s'adapter à toute extension et modification.

Ces expériences ont également montré la nécessité de mettre en œuvre des interfaces intelligentes, adaptables en fonction de l'utilisateur. C'est-à-dire développer des systèmes « boîte noire dans une boîte de verre » *(a black box in a glass box)* où seuls les niveaux conceptuels les plus élevés sont accessibles à l'utilisateur, la complexité linguistique restant cachée. En effet le système présente à l'utilisateur des classes de messages identifiables plutôt qu'un ensemble de propriétés linguistiques complexes et ingérables.

9. Conclusion

Cet article propose une approche évolutive qui s'adapte à la nature des messages au cours du temps et qui exploite le maximum d'informations pour filtrer le courrier électronique. Ce travail nous a permis de valider deux hypothèses principales.

– Les études sur corpus nous semblent un passage obligé dans la conception d'un système de filtrage automatique. En effet, nous avons examiné une application d'un principe d'analyse automatisée reposant sur des traitements linguistiques faibles au problème du filtrage d'information. Nous avons tenté de déterminer la relation et l'adéquation entre discrimination textuelle et occurrence de propriétés linguistiques. Nous avons donc montré l'apport d'une étude linguistique des corpus dans un domaine applicatif. Toutefois, les résultats acceptables enregistrés dans notre expérience ne doivent pas occulter le fait que l'adéquation entre propriétés linguistiques et types de textes n'est pas parfaite.

– Les ressources linguistiques et les différents outils et méthodes issus du TAL constituent une source importante d'amélioration de la qualité des systèmes d'analyse automatique et plus particulièrement de filtrage d'information permettant en particulier d'améliorer la représentation de l'information et d'offrir ainsi des performances supérieures.

Dans le domaine du filtrage, il faut considérer le problème des variations linguistiques. En effet, les mots-clés (multiples) composant les profils n'apparaissent souvent pas de façon littérale dans le message. Le traitement de la variation linguistique des mots permet d'introduire une flexibilité dans la procédure

d'appariement entre le profil et le message à filtrer. En effet, des réalisations linguistiques différentes portant le même contenu informationnel peuvent être regroupées et considérées comme équivalentes.

Des outils tels que les traitements linguistiques et les ressources existantes ou acquises sur corpus (exemple la lemmatisation) autorisent la prise en compte de variantes et peuvent donc aider à la désambiguïsation des mots, améliorer la représentation textuelle et par conséquent les performances.

En plus de la variation linguistique, notre approche fait appel à des propriétés linguistiques qui peuvent aider à améliorer les résultats de filtrage. En effet, les propriétés matérielles, énonciatives, structurelles et syntaxiques permettent à notre système, contrairement aux systèmes existants, de relier un message à un profil même s'ils n'ont pas de mots-clés en commun (propriétés lexicales). Ce qui permet d'augmenter le taux de *rappel* tout en gardant une bonne *précision*.

Ainsi, à travers notre expérience, les méthodes linguistiques combinées aux méthodes statistiques semblent prometteuses pour générer un filtrage efficace de l'information sur les réseaux de communication.

10. Bibliographie

Aït-Mokhtar S., Chanod J.-P., «Xerox Incremental Parser (XIP)», *Proceedings of the Fifth Conference on Applied Natural Language Processing*, 1997, p. 72-79.

Amini M. R., Apprentissage automatique et recherche de l'information : application à l'extraction d'information de surface et au résumé de texte, Thèse de doctorat, Université de Paris 6, 2001.

Androutsopoulos I., Koutsias J., Chandrinos K.V., Paliouras G., Spyropoulos C.D, «An evaluation of naïve Bayesian anti-spam filtering», *11 th European Conference on Machine Learning (ECML 2000)*, Barcelona, Spain, p. 9-17.

Apté C., Damerau F., Weiss S., «Automated Learning of Decision Rules for Text Categorization», *ACM Transactions on Information Systems*, vol. 12, n° 3, p. 233-251.

Benveniste E., *Problème de linguistique générale,* Gallimard, Paris, 1966, p. 251-257.

Biber D., *Variation Across Speech and Writing,* University Press, Cambridge, 1988.

Bronckart J. P., Bain D., Schneuwly B., Davaud C., Pasquier A., *Le fonctionnement des discours : un modèle psychologique et une méthode d'analyse*, Lausanne, Delachaux & Niestlé, 1985.

Bellot P., El-Bèze M., « Classification locale non supervisée pour la recherche documentaire, Traitement Automatique des Langues », *TAL 2001,* vol. 42, n° 2. janvier 2001, Hermès, p. 335-366.

Ben Hazez S., Desclés J. P., Minel, J.L., « Modèle d'exploration contextuelle pour l'analyse sémantique des textes », *TALN 2001*, Tours, p. 73-82.

Brill E., «A Simple Rule-based Part of Speech Tagger», *Proceedings of the Third Conference on Applied Natural Language Processing*, *ACL*, 1992, p. 152-155.

Caropreso M., Matwin S., Sebastiani F., «A learner-independant evaluation of the usefulness of statistical phrases for automatic text categorization», *Text Databases and Document Management: Theory and Practice*, A.G. Chin ed., Idea Group Publishing, 2001, p. 78-102.

Carreras X., Marquez L., «Boosting Trees for Anti-Spam Email Filtering», *Proceedings of RANLP-01, 4th International Conference on Recent Advances in Natural Language Processing,* 2001.

Chandrasekar R., Srinivas B., «Using Syntactic Information in Document Filtering : A Comparative Study of Part-of-speech Tagging and Supertagging», *Proceedings of the RIAO-97 Conference*, 1997, p. 531-545.

Collins M. J., «A New Statistical Parser Based on Bigram Lexical Dependencies», *Proceedings of the 34th Annual Meeting of the ACL*, June 1996, Santa Cruz, CA.

Copeck T., Barker K., Delisle S., Szpakowicz S., «Automating the Measurement of Linguistic Features to Help Classify Texts as Technical», *Conference TALN 2000*, Lausanne, 2000.

Davalo E., Naim P., *Des réseaux de neurones,* Eyrolles, 1993.

Desclés J. P., Cartier E., Jackiewicz A, Minel J. L., «Textual Processing and Contextual Exploration Method», *CONTEXT'97*, Rio de Janeiro, Brésil, 1997, p. 189-197.

Dreyfus G., Martinez J.M., Samuelides M., Gordon M.B., Badran F., Thiria S., Hérault L., *Réseaux de neurones méthodologies et applications*, Eyrolles, 2002.

Garcia D., « Exploitation pour l'élaboration de requêtes de filtrage de texte, des connaissances causales détecté par COATIS », *RIFRA'98 Rencontre internationale sur l'extraction, le filtrage et le résumé automatique*, 1998, p. 44-54.

Habert B., Illouz G., Lafon P., Fleury S., Folch H., Heiden S., Prévost S., « Profilage de textes : cadre de travail et expérience », *JADT2000 : 5ème Journées Internationales d'Analyse Statistique des Données Textuelles*, 2000.

Joachims T., «A probabilistic analysis of the rocchio algorithm with tfidf for text categorization», *Proceedings of ICML-97, 14th International Conference on Machine Learning*, 1997, p. 143-151.

Joachims T., «Text categorization with support vector machines : learning with many relevant features», *Proceeding of ECML-99, 16th European Conference on Machine Learning*, 1999, p. 137-142.

Junker M., Abecker A., «Exploiting Thesaurus Knowledge in Rule Induction for Text Classification», *Proceedings of the RANLP-97 Conference,* 1997, p. 202-207.

Lewis D.D., «An evaluation of phrasal and clustered representations on a text categorization task», Proceedings of SIGIR-92, *15th ACM International Conference on Research and Development in Information Retrieval*, Copenhagen, Denmark, 1992, p. 35-50.

Lewis D.D., Ringuette M, «Comparison of two learning algorithms for text categorization», P*roceedings of the Third Annual Symposium on Document Analysis and Information Retrieval SDAIR'94,* 1994.

Marcu D., « From discourse structures to text summaries », *Workshop Intelligent Scalable Text Summarization,* Madrid, Spain, 1997.

Minel J. L., Desclés J. P., Cartier E., Crispino G., Ben Hazez S., Jackiewicz A., « Résumé automatique par filtrage sémantique d'informations dans des textes, Présentation de la plate-forme FilText », *Revue Technique et Science Informatique,* n° 3, 2001.

Mc Callum A., Nigam K., « A comparison of event models for naïve Bayes Text classification », *Learning for text categorization*, 1998.

Miller G., « WordNet : An On-line Lexical Database », *International Journal of Lexicography*, 1990.

Namer F., « Flemm : Un analyseur flexionnel du français à base de règles », *Traitement automatique des langues pour la recherche d'information, TAL,* 2000, vol. 41, n°2, Paris, p. 523-547.

Orasan C., Krishnamurthy R., «A corpus-based investigation of junk emails», *Proceedings of LREC-2002*, Las Palmas, Spain, 2002.

Poibeau T., Nazarenko A., « L'extraction d'information, une nouvelle conception de la compréhension de texte ? », *TAL*, vol. 40, n°2, 1999, p. 87-115.

Sahami M., Dumais S., Heckerman D., Horvitz E., «A Bayesian approach to filtering junk e-mail», *Learning for Text Categorization Papers from the AAAI Workshop,* Madison Wisconsin, 1998, p. 55-62.

Salton G., Wong A., Yang C., «A vector space model for information retrieval», *Communications of the ACM*, vol. 18, n° 11, 1975, p. 613–620.

Sebastiani F., « A Tutorial on Automated Text Categorisation », *Proceedings of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence*, 1999.

Virbel J., Luc Ch., « Le modèle d'architecture textuelle : fondements et expérimentation », *Verbum*, vol. 23, n° 1, 2001, p. 103-123.

Yang Y., Pedersen J. O., « A comparative Study on Feature Selection in Text Categorization », *International Conference on Machine Learning ICML1997,* Nashville, TN, USA, 1997.